The Implementation of Image Reconstruction Algorithm for Liner Sensor

Woong-Sik Kim

Department of Medical IT Engineering Konyang University
#119 Daehak-ro, Nonsan, Chungnam, South Korea

wskim@konyang.ac.kr

Abstract- **Linear Sensor performs a series of snapshots capturing partial images of the object. We assume non-uniform objects' motion across the sensing area. Snapshot rate is variable depending on characteristics of the object's motion, the algorithm, which is described here, defines adaptively this rate together with characteristics of the object's motion. Algorithm defines a full set of the coordinate parameters (x, y shift and rotation) for each of sequentially captured partial images (later - stripes). Those coordinate parameters are used to provide proper placement of strips into the area of the final image and to reconstruct the image. Algorithm provides additional information about current status of image reconstruction process, mechanism to identify start image, end of image and synchronization loss conditions. Algorithm provides mechanismwith adaptive capturing rate changes.**

Keywords-Linear sensor; Fingerprint; Recognition; Image Processing; Reconstruction

I. INTRODUCTION

The linear sensor produces narrow sequential partially overlapped grayscale image stripes with the rate adjustable by the algorithm[1]. Each stripe has at least several rows of discrete dots or pixels[2],[3]. Vertical sweeping speed is regular enough, so we can extrapolate it and be able to estimate, adjust the rate to the proper value[4].

We assume that two sequential stripes may have very close horizontal coordinates and rotation angles, so in some cases we may avoid taking into account those parameter's estimations[5].

The task is to perform on-fly and on-board image reconstruction. Reconstructed image data transferred to the computer or to the matching board might be performed as the parallel task, while the rest of the image reconstruction is in process[6].

Under the term "Linear Sensor" we assume that one dimension of the sensor's area in terms of dots is reasonably less than another (the term "reasonably less" will be defined later), we will call column the line of dots in a shortest dimension and rows line of dots in a longest dimension direction.

We assume that sensor has at least several rows[7]-[9].

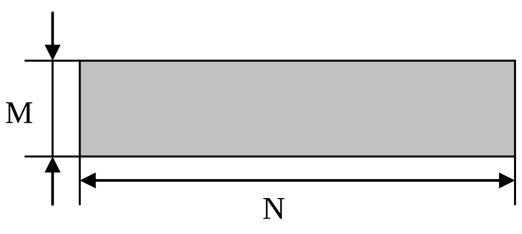


Fig. 1 Linear sensor area

For the description purposes we assume horizontal orientation of sensor's rows and zero angle direction of the object motion from left to right(x axis), we will call this parallel direction, relatively to the sensor orientation, and perpendicular direction(y axis).

We will call tangential, relative to the sensor orientation. We will use small letters (x , y , α) for the sensors' coordinate system and capital letters (X, Y, A) for the objects' coordinate system.

We use prefix Δ to identify coordinates differences: Δx, Δy , $\Delta\alpha$. We will define N as the amount of columns on sensor's area and M as the amount of rows in dots(F. 1).

We assume "N is reasonably more than M" as N$\geqslant$M$^{2.}$

Captured stripes have same dimensions as the sensor's area. Under term "object motions across the sensor" we assume both: object motion across the sensor and sensor motion across the object.

II. IMAGE RECONSTRUCTION PROCESS

Image Reconstruction Process consists of series of steps by amount of strips captured. Strip-to-strip comparison is performed on each step to estimate relative coordinate changes: tangential, parallel shifts of the strip middle and rotation angle (F. 2).

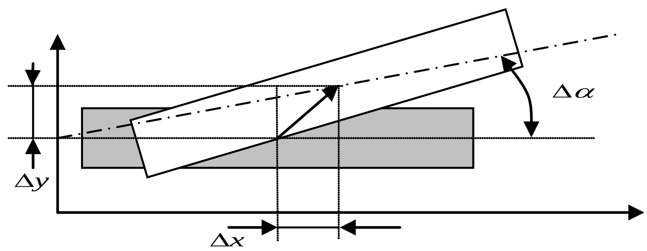


Fig. 2 Middle and rotation angle

All three coordinate parameters are independent. Parameters are taken in local coordinate system of earlier captured strip.

To perform comparison overlapping between two strips must exist. To achieve the overlapping capture rate is adaptively tuned according to the object's speed changes. Condition for capture rate change is that at least one of the coordinate parameters went far away from predefined limit values. Proposed limit values for each of the coordinates are:

$$|\Delta x|_{opt} \approx \frac{N}{2M},\ |\Delta y|_{opt} \approx \frac{M}{2},\ |\Delta\alpha|_{opt} \approx \frac{M}{2N} \qquad (1)$$

This will provide $\approx 50\%$ overlapping of the strips. Sensor's capturing rate v_{i+1} for the next step can be estimated from one previous capturing and comparison step as:

$$v_{i+1} = \max\left(\frac{2v_i\Delta y_i}{M}, \frac{2Mv_i\Delta x_i}{N}, \frac{2Mv_i\Delta\alpha_i}{N}\right) \quad (2)$$

It assumes that object speed will not change much between two consecutive captures. In case object acceleration is assumed high for better prediction, extrapolation of $v\Delta y$, $v\Delta x$, $v\Delta\alpha$ might be used for the several previous steps.

In case of synchronization loss (no overlapping between stripes obtained), but image is present, maximum possible rate used to get back the synchronization and continue image capture process.

Rotation is not used for strips' comparison. By taking into account that the angle between two stripes is kept small (because of requirement of strips overlapping and adaptive capturing rate) rotation is substituted by skew, which is much preferable for calculation purposes, we (according to 1, 2) will have sub-pixel accuracy of this approximation if it will operate with at least M separate segments (e.q. F. 3), where M is the same value as amount of sensor's rows.

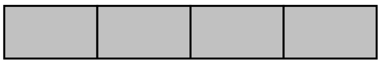


Fig. 3 M separate segments

One strip's segment is compared to another strip's corresponding segment independently from other segment pairs. Local (x, y) shifts values Δx_n and Δy_n are produced as the locations of the different function minimums for each segment pairs.

Mean squire linear approximation with weight coefficients is used to estimate the slope $\Delta\alpha$, tangential Δy and parallel Δx shift values. Reversed difference functions minimum values are used as weight coefficients for each of M segment pairs.

Two separate equidistant approximations are performed to obtain the slope of $\Delta x_n = \Delta x(n)$ and $\Delta y_n = \Delta y(n)$ dependencies and Δx, Δy values (F. 4).

Shifts Δy and shift Δx are obtained as the approximation function value in the middle of initial strip. The slope of the next strip is calculated as:

$$tg(\Delta\alpha) = \frac{M * tg(\alpha_y)}{N + M * tg(\alpha_x)} \quad (3)$$

Graphical interpretation of this result is shown on F. 5.

This gives us three parameters of the object's motion at the time between two consecutive image captures: tangential shift, parallel shift and the rotation angle.

This set of parameters in local (sensor's) coordinate system completely defines relative position of one strip compare to another and let us to reconstruct the image of the object passed through the sensing area.

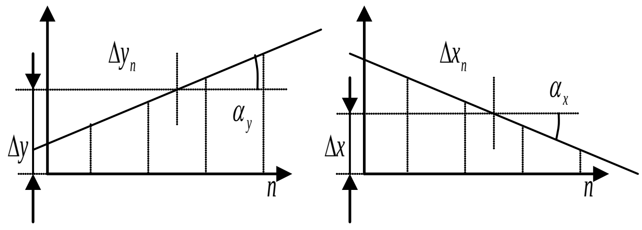


Fig. 4 Δx, Δy Value

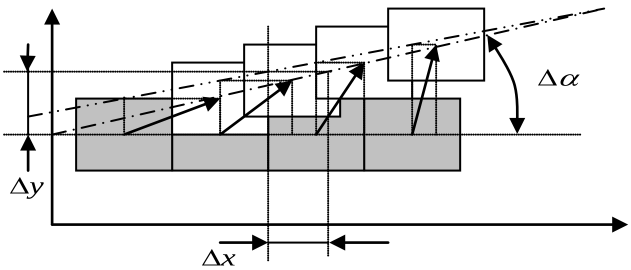
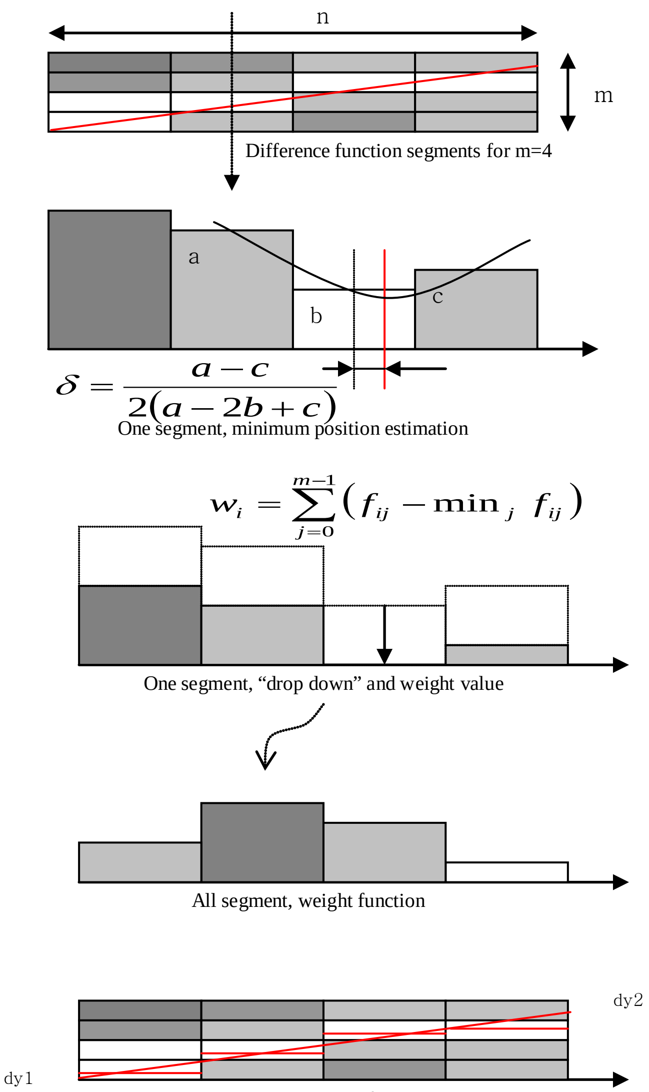


Fig. 5 Graphical interpretation of this Result



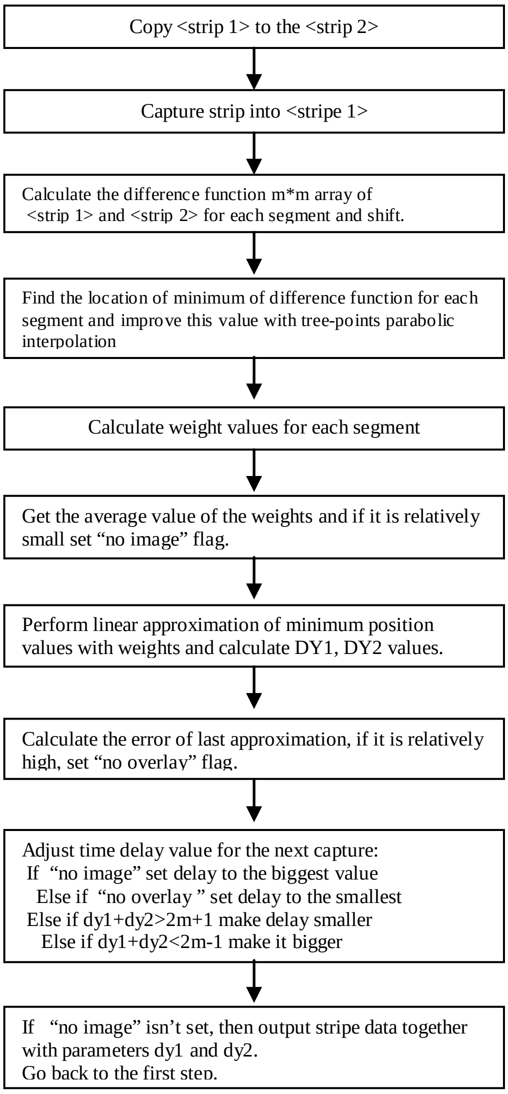


Fig. 6 Demonstration purposes of object motion

For demonstration purposes several kinds of object motions across the sensor is shown in F. 6.

Case-a is straight vertical motion, in case-b sensor and object motion direction is not perpendicular and as the result we have sideway component Δx case-c is combined with vertical motion and rotation, while case-d is similar to case-c.

We have horizontal shift in image coordinates ΔX just because of different rotation center locations. In case-e we are unable to reconstruct image, because not all parts of the object pass the sensing area.

Values X, Y, A completely define the placement of the current stripe in global image coordinates. Coordinate translation from local (sensor’s) to global(image’s) is performed by recursion procedure:

$$A_{i+1} = A_i + \Delta\alpha,$$
$$X_{i+1} = X_i + \Delta x \cos(A_{i+1}) - \Delta y \sin(A_{i+1}),$$
$$Y_{i+1} = Y_i + \Delta x \sin(A_{i+1}) + \Delta y \cos(A_{i+1}) \qquad (4)$$

III. EXPERIMENTS

We proposed the technology which we can reconstruct the perfect image no matter the input image with various directions, vertical, horizontal or rotating. With this proposed technologies, we can reconstruct the image from estimation and compensation the scanned image through Fingerprint Sensor. And the recognition rate can be highly improved.

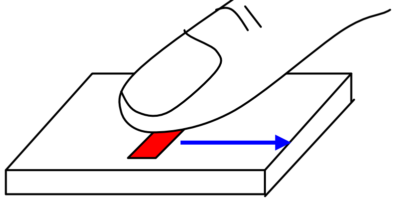


Fig. 7 Object moving of linear sensor

Term Model is used to indicate that this application simulates the linear sensor image capture process and assume some variations in sweeping object behavior, like speed or rotations.

Also Model application panel provides the visualization of the process and provide the user with the controls to check the other applications for the variations of speed and rotations and even for some “sideway jitter”.

Linear sensor and strip comparison principle assume partial stripes overlapping, so in some extreme cases, when there is not such overlapping or the area of it is too small, or the data in overlapping area is “flat”, there is no way to get the correct coordinate values and to place this stripe into the proper place of entire image. Partially to overcome that “Estimator” algorithm uses “No overlapping” check and proper reset after such an event. Also big angles in most of the cases just cannot be estimated because of small stripe overlapping at this case.

Linear Type Sensor requires the reconstructing technologies with estimating the speed of finer object on the sensor.

We have tested the algorithm with ‘Reconstructor’ which is generating the one combined image and ‘Estimator’ estimating the speed of constant outputs from sensor.

Fig. 7 shows the reconstructed image using stripes acquired from linear sensor by using vertical image estimation only, which is the image obtained using existing algorithms.

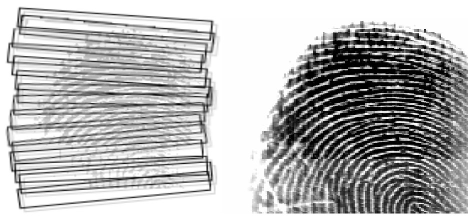


Fig. 8 Reconstructed fingerprint image through proposed algorithm

Fig. 8 shows the reconstructed image without any distortion using the proposed method that estimates vertically, horizontally, and rotationally.

Fig. 9

Fig. 10

Fig. 9 shows the reconstructed image using stripes acquired from linear sensor by using vertical image estimation only, which is the image obtained using existing algorithms.

Fig. 10 shows the reconstructed image without any distortion using the proposed method that estimates vertically, horizontally, and rotationally.

Fig. 11 Reconstructed fingerprint image through proposed algorithm

The below Fig. 11 is example for captured fingerprint image on 515dpi through our algorithm.

IV. CONCLUSIONS

Most of linear sensors have inferior image restoration than the general optical sensor due to the skill of user and negligence. However, with the price and slim shape of the sensors, it has been applied in several fields such as mobile phone and others, and if the ensuing research could improve the image restoration for even more, there would be slim shape and economic benefit that contributes to the development of bio-recognition industry.

On a final stage tests must be performed to make a conclusion about FAR/FRR values. Those tests might be performed on a large fingerprint database. Optimizing, speed and memory estimation need to be performed for the hardware implementation (matching board). Depending on the desired fingerprint database size, different algorithms might be used to achieve the balance between FAR/FRR, speed and memory requirements.

REFERENCES

[1] Woong-Sik Kim, Weon-Hee Yoo, “A Method for Acquiring Fingerprint by Linear Sensor,” Lecture Notes in Computer Science, Volume 3684, Aug 2005, Pages 410 – 416

[2] A.K.Jain, L. Hong, R. Bolle, “On-Line Fingerprint Verification”, IEEE Trans., Pattern Analysis and Machine Intelligence, vol. 19,no. 4,pp. 302-313, April., 1997.

[3] A.K. Jain, S.Prabhakar, L. Hong, and S. Pankanti, “Filterbank-based fingerprint matching”, IEEE Trans. On Image Processing. 9(5):846-859, May 2000.

[4] J. D. Stoszand L. A. Alyea, "Automated Systems for Fingerprint Authentication Using Pores and Ridge Structure," Proceedings of SPIE, Automatic Systems for the Identification and Inspection of Humans (SPIE Vol. 227 7), San Diego, 1994, p. 210- 223.

[5] Dario Mario, and David Maltoni, “Direct Gray-Scale Minutiae Detection In Fingerprints”, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 1, , pp. 27-39, Jan., 1997.

[6] Lin Hong, Yifei Wan, and Anil Jain, “Fingerprint Image Enhancement: Algorithm and Performance Evaluation”, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20,no. 8,pp. 777-789,Aug., 1998.

[7] Nalini K, and A.K. Jain, “A Real-Time Matching System for Large Fingerprint Database”, IEEE Trans., Pattern Analysis and Machine Intelligence, vol. 18,no. 18,pp. 799-813, Aug., 1996.

[8] Xudong Jiang and Wei-Yun Yau, “Fingerprint Minutiae Matching Based on the Local and Global Structures”, International conference on pattern Recognition(ICPR’00)-Vol 2.,no. 18,pp. 1038-1042, Sep., 2000.

[9] T. MAEDA, M. Matsushida and K. Sasakawa, “Identification Algorithm Using a Matching Score Matrix”, IEICE Trans. INF. & SYST., Vol.E-84-D,no. 7, pp. 819-824, July 2001.

[10] Xudong Luo, Jie Tian and Yan Wu, “A Minutia Matching algorithm in Fingerprint Verification”, International conference on pattern Recognition (ICPR’00)-Vol 4., no. 18,pp. 4833-4841, Sep., 2000.

[11] Woong-Sik Kim, Weon-Hee Yoo, “A Layered Fingerprint Method,” Lecture Notes in Computer Science, Volume 3546, July 2005, Pages 702 – 709